LABORATORIO DI LINGUAGGI-A
Docente: András Horváth
Prova scritta del 31/04/2005

COGNOME _____

NOME _____

MATRICOLA _____

Rispondere alle domande negli spazi sottostanti (o sul retro del foglio). **Non è consentito usare altri fogli.**

1. (3 punti) Cosa stampa a video il seguente programma?

Soluzione ↓

```
#include<stdio.h>

int * f(int *a, int *b)
{
  int i1;

  for(i1=0;i1<2;i1++) b[i1]=a[i1];
  b[1]*=2;
  b[2]+=3;
  return a-1;
}

void main(void)
{
  int a[]={5,4,3,2,1},*b;

  b=f(a+1,a);
  printf("%d %d %d\n", b[0],b[1],b[2]);
}
```

2. (3 punti) Cosa stampa a video il seguente programma?

Soluzione ↓

```
#include<stdio.h>

void main(void)
{
  int i1=6,i2;
  do {
    i2=1;
    while(i2++<i1)
      printf("%d ",i1*i2);
    printf("\n");
  } while(i1-->=3);
}
```

3. (4 punti) Cosa stampa a video il seguente programma?

Soluzione ↓

```
#include<stdio.h>

void main(void)
{
  char stringa[]="luna";
  char *stringhe[]=
    {"mamma", "bella", "ciao", "casa"};

  printf("%c\n", stringa[1]);
  printf("%c\n", *(stringa+3));
  printf("%s\n", stringa+1);
  printf("%s\n", &stringa[3]);
  printf("%s\n", stringhe[2]);
  printf("%s\n", stringhe[3]+1);
}
```

4. (3 punti) Cosa stampa a video il seguente programma?

Soluzione ↓

```
#include<stdio.h>

int fun1(int a)
{
  if(a<4) return a;
  return fun1(a-2);
}

int fun2(int a)
{
  if(a>8) return a;
  return a+fun1(a);
}

void main(void)
{
  printf("%d %d %d %d\n",
         fun1(10), fun2(8), fun2(4), fun1(1));
}
```

5. (2 punti) Scrivere delle espressioni logiche in C che corrispondono ai seguenti enunciati.

Soluzione ↓

a) `a` non è uguale a `b` ed è maggiore di `c`

b) il cubo di `a` è minore di `c`

c) `a` è uguale a `b` o `b` non è uguale a `5`

d) il terzo carattere della stringa puntata da `s` è un numero

6. (5 punti)
Scrivere una versione **ricorsiva** di `strcmp(char *s1, char *s2)` che restituisce

- `-1` se la stringa puntata da `s1` è lessicograficamente precedente alla stringa puntata da `s2`,
- `0` se la stringa puntata da `s1` è identica alla stringa puntata da `s2`,
- `+1` se la stringa puntata da `s1` è lessicograficamente successiva alla stringa puntata da `s2`.

7. (5 punti) Scrivere una funzione che

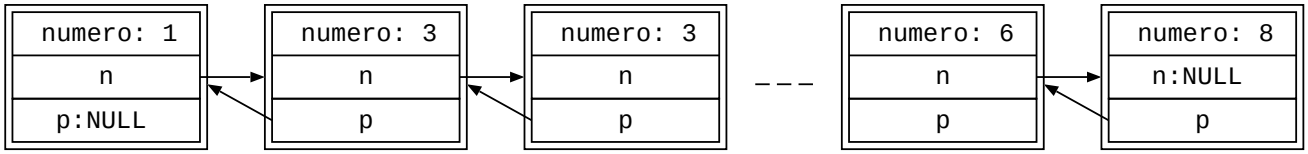
- prende due numeri interi, r e c
- e restituisce un puntatore che punta ad un vettore di puntatori di tipo `int`.

Il puntatore restituito è usato per accedere agli elementi di una matrice rettangolare con r righe e c colonne. Il j-esimo elemento della i-esima riga della matrice deve contenere $i + j$ se $i \leq j$ altrimenti $i \times j$. (La funzione deve allocare memoria per la matrice.)

8. (7 punti) La seguente struttura è definita per memorizzare una serie di numeri in ordine **non decrescente**:

```
struct node {
  int numero;
  struct node *n
  struct node *p;
};
```

Un esempio dell'uso della struttura è riportato nella seguente figura:



Scrivere una funzione che ha il seguente parametro

- un puntatore di tipo `struct node` che punta al primo elemento della lista

e stampa a video tutti i numeri della catena in ordine **non crescente**. La funzione non deve restituire nessun valore.

Scrivere una funzione che ha i seguenti parametri

- un puntatore di tipo `struct node` che punta al primo elemento della lista
- un numero di tipo `int`

e cancella un nodo della lista che contiene il numero passato (se esiste un tale nodo). La funzione deve restituire un puntatore di tipo `struct node` che punta al primo elemento della lista.